基于序列比对算法的中文文本相似度计算研究*

■ 赵登鹏[1] 熊回香[1] 田丰收[2] 李昕然[1]
[1] 华中师范大学信息管理学院 武汉 430079 [2] 高寻真源教育科技有限公司技术研发部 济南 250000

摘 要：[目的/意义]针对序列比对算法在文本相似度中的应用，改进全局比对算法并提高该算法的准确性，同时，应用局部比对算法有效解决内容差异或长短差异较大的两文本进行比对的问题。[方法/过程]首先，利用 HanLP 中的 CRF 模型对在线学术资源中文文本数据集进行规范化处理，构成中文序列集；然后，使用最新的中文维基百科语料训练 Word2Vec 模型来构建语词对打分矩阵；最后，基于打分矩阵和改进的打分规则，对进行全局比对/局部比对的两中文序列进行比对并获得比对的最优解，回溯该最优解，获取最优解的比对路径，计算两中文序列的相似度。[结果/结论]实验结果表明，相较于目前全局比对算法的相关研究，本文基于词性标注的结果与 Word2Vec 构建的语词对打分矩阵进一步提升了全局比对算法计算文本相似度的准确性，同时，应用于文本相似度计算的局部比对算法能够有效解决内容差异或长短差异较大的两文本进行比对的问题。

关键词：CRF 模型 词性标注 Word2Vec 序列比对 局部比对 文本相似度
分类号：TP391.1
DOI：10.13266/j.issn.0252-3116.2021.11.011

1 引言

随着信息技术的迅速发展，对互联网产生的海量文本信息进行挖掘和研究能提供给用户有价值的内容，如文本的分类聚类、个性化推荐、信息抽取、信息检索、搜索引擎等，而文本相似度作为衡量文本间的差异和共性的方法，也是这些技术任务的核心环节[1-2]。近年来，文本相似度主要被应用在词义消歧、自动摘要抽取、机器翻译自动评估、数据库的模式匹配及语义异构问题等研究中[3]。在中文信息处理领域，计算中文字符串，如词语、词组等的相似度计算对词典编纂、基于实例的机器翻译、自动问答、信息过滤等都具有重要的作用[4]，目前，文本相似度计算领域主要包含了基于字符串的方法、基于语料库的方法、基于知识库的方法和混合方法[5-10]，其中序列比对算法属于基于字符串的方法且该方法用于时序数据和流式数据具有不错的效果[11]。此外，序列比对算法在中文里的应用根据所比对字符粒度大小和比对方式的不同还能用于语义挖掘、文本分类与聚类、个性化推荐、智能检索等。

序列比对算法源于生物信息学领域，是对序列进行分析从而了解基因结构和功能最常用和最经典的研究手段，通常是对氨基酸序列之间或核酸序列之间两两比对来比较两条序列之间的相似区域和保守性位点寻求同源结构，揭示生物进化、遗传和变异等问题[12]。1970 年，S. B. Needleman 与 C. D. Wunsch 提出了双序列全局比对算法[13]；1975 年，T. F. Smith 与 M. S. Waterman 在 S. B. Needleman 与 C. D. Wunsch 所提出算法的基础上提出了改进的双序列局部比对算法[14]；之后，随着生物信息学的不断发展，2019 年，出现了诸多序列比对的工具及软件[15-19]并不断改进完善，近年来关于序列比对算法的研究多是对于序列比对算法的改进与加速[20-21]，同时，2020 年 R. J. LU，X. ZHAO 等还使用序列比对算法来研究了 COVID-19 与 SARS 病毒、MERS 病毒的基因相似性[22]。在图情领域，2010 年，徐硕等[23]最先提出使用全局比对算法来计算中文文本的相似度，解决了传统的语义相似度计算方法没有考虑文本语词顺序的问题；2014 年，王汀等[24]提出的全局比对算法中，参考田久乐等[25]对于同义词林的研究，改进了全局比对算法比对词语的合理性与准确性，但受限于同义词林的覆盖范围和广度，该方法只能在

* 本文系国家社会科学基金项目“融合知识图谱和深度学习的在线学术资源挖掘与推荐研究”（项目编号：19BTQ005）研究成果之一。
作者简介：赵登鹏（ORCID：0000-0002-7699-5222），硕士研究生，E-mail：1251508909@qq.com；熊回香（ORCID：0000-0001-9956-3396），教授，博士生导师；田丰收（ORCID：0000-0001-8789-4032），硕士研究生；李昕然（ORCID：0000-0002-3134-9876），硕士研究生。
收稿日期：2020-12-09 修回日期：2021-02-24 本文起止页码：101-112 本文责任编辑：易飞

特定领域有一定效果；熊回香等[26]基于 Word2Vec 来构建语词对打分矩阵，大大提高了该算法的准确性并有效处理了中文文本中所出现的“重复词对”的问题。

但目前的相关研究中，对于序列比对算法的研究还存在一个局限，即只有当分词后的两条中文文本之间在内容和长度上差异较小时，全局比对算法才较为有效，针对这一情况，本文提出了改进的局部比对算法来应对内容和长度差异较大的两文本进行序列比对的问题。为了更好地将序列比对算法运用到中文文本相似度计算研究当中，本文基于 CRF 模型词性标注的结果与 Word2Vec 构建的语词对打分矩阵来进一步提高全局比对算法以更好地挖掘中文文本之间的相似性关系，同时，应用局部比对算法来有效解决内容差异或长短差异较大的两本文进行比对的问题，进一步提升序列比对算法计算中文文本相似度的效果与准确性，以使得序列比对算法在中文文本相似度中的运用更为准确合理。

2 CRF 模型、Word2Vec 与序列比对算法

2.1 CRF 模型

随着研究者不断提出各种应用于语言信息处理领域的数字模型，基于统计的分词技术逐渐成为主流，HMM、MEMM 以及 CRF 是常被用到的 3 种统计模型[27]。其中 CRF 没有 HMM 那样严格的独立性假设条件，可以更好地容纳上下文信息；同时，CRF 模型具有 MEMM 判别式模型的特点，对数据量要求小、速度快、准确率高，相比较传统的一些分词模型与工具，具有一定的优势，并常被用于句法分析、命名实体识别、词性标注等自然语言处理任务当中。

HanLP 是由何晗 2014 年开发并开源于 GitHub 的一款 NLP 工具，其包含了最长匹配、HMM、感知机、CRF 等自然语言处理模型，同时，HanLP 参考 CoNLL-X[28]、Biaffine[29]、FastText[30]、BERT[31]等更进一步提高了 HMM、CRF 等处理自然语言处理的效果和准确性。HanLP 所包含的 CRF 模型具备了良好的分词与词性标注功能，也能很好地识别未登陆词，基于此，笔者选取 HanLP 的 CRF 模型来完成实证研究中一系列的自然语言处理任务。

2.2 Word2Vec 模型

Word2Vec 是 Google 于 2013 年以深度学习的思想为基础开发的一种词向量模型，主要用于实现文本信息由非结构化形式到向量化形式的转变[32]。自 Word2Vec 发布以来，Word2Vec 已在自然语言处理领域得到了广泛的应用，以其为基础进行的各种研究也在逐步递增，Word2Vec 目前已成为自然语言处理领域最具代表性的工具之一。Word2Vec 通过学习文本能够将字词转换为向量的形式，并用词向量的方式表征词的语义信息[33]。此外，Word2Vec 作为一种自然语言处理工具，其最大的特点之一就是以上下文信息为基础实现词的特征表示，从而解决维度灾难的问题。

2.3 序列比对算法

序列比对算法主要分为 2 种，即寻找序列之间全局相似性的全局比对算法与寻找序列之间局部相似性的局部比对算法，两种算法共用同一语词对打分矩阵来比对文本中具有相似关系的词语，以进一步探究文本之间的相似关系。

2.3.1 相关概念基础

序列比对算法应用于中文文本相似度的研究，是将两个中文文本分词后处理为以词语形式按顺序排列的两条中文序列，然后将两条中文序列排列在一起比较其相似之处，序列中可通过插入空位符以使得两条序列中尽可能多的相同或相似的词语排在同一列上。为更好地阐明序列比对算法如何用于研究中文文本，参考文献[12][26]对本文方法涉及的相关概念进行阐述：

（1）中文序列 $cs_i(i \in \{1,2,3,\cdots,n\})$。$cs_i$ 是某一中文文本经过预处理、分词而获得的语词序列，且形式化表示为 $cs_i=\{t_{i,1},t_{i,2},\cdots,t_{i,k},\cdots,t_{i,m}\}$，其中 $t_{i,k}$ 表示 cs_i 第 k 个词语，这些语词按照原来的顺序依次排列构成 cs_i。

（2）中文序列集 CS(Chinese Sequence Set)。CS = $\{cs_1,cs_2,\cdots,cs_i,\cdots,cs_n\}$，n 表示 CS 中所含中文序列的个数，$cs_i$ 表示中文序列集 CS 中的第 i 条中文序列。

（3）比对矩阵 M(Alignment Matrix)。$M_{m\times q}$ 表示 $cs_i=\{t_{i,1},t_{i,2},\cdots,t_{i,k},\cdots,t_{i,m}\}$ 与 $cs_j=\{t_{j,1},t_{j,2},\cdots,t_{j,p},\cdots,t_{j,q}\}$ 进行比对的过程中所产生的结果，由于在进行比对之前要在该矩阵中插入一行空位符“－”和一列空位符“－”，最终 M 的大小为 $(m+1)\times(q+1)$，其中 $M_{k,p}$ 表示 cs_i 的第 k 个词语与 cs_j 的第 p 个词语进行比对的结果。

（4）语词对打分矩阵 W(Words Grade Matrix)。若要对 cs_i 与 cs_j 进行比对，就需要有一个介于 0－1 之间的值来度量任意两语词 $t_{i,k}$ 与 $t_{j,p}$ 的相似性，将该值记为 $sim(t_{i,k},t_{k,p})$，语词对打分矩阵则用来保存所有满足条件的 $sim(t_{i,k},t_{k,p})$，以供给 cs_i 与 cs_j 进行比对时参考。

（5）打分规则 G(Grade Rules)。cs_i 与 cs_j 进行比

对时，若所比对的两词语出现错位匹配或空位匹配就惩罚0.05分(即G = －0.05分)；若所比对的两词语能够参考W获得一个0－1之间的分值，则是相似匹配；若所比对的两词语完全相同则是完全匹配，此时G = 1分。

(6)比对得分的计算S(Final Alignment Scores)。cs_i与cs_j的比对结束后，匹配后的每一对词语此时都有了一个打分，根据不同的序列比对算法的要求，将有效的打分进行累加，就能得到比对结果最终的得分S。

序列比对算法之所以能很好地应用于生物信息学领域，是因为其参考了基于海量核酸以及氨基酸统计构建的打分矩阵，Word2Vec基于上下文环境相似的两个词有着近似含义的思想，经过大量语料训练之后，可以很好地表示出语词的词向量并通过计算向量余弦值来量化语词对在数值上的关系，这种方式与生物信息学领域构建核酸与蛋白质的打分矩阵的思想十分接近。因此，为促使序列比对算法能够更加合理有效地运用于中文文本相似度计算的研究当中，本文选取了Word2Vec来计算中文词语之间的相似性以构建用于中文文本比对所需的语词对打分矩阵。

上述相关概念中，语词对打分矩阵则是序列比对算法应用于研究中文文本之间相似性的核心基础，该打分矩阵供给cs_i与cs_j进行比对时所参考，从而更好地度量cs_i与cs_j的相似关系。构建打分矩阵的过程如图1所示，使用训练好的Word2Vec来计算cs_i与cs_j之间任意两词语的余弦相似度，并将所有满足打分条件的$sim(t_{i,k},t_{k,p})$放入语词对打分矩阵当中后，就构建好了打分矩阵。

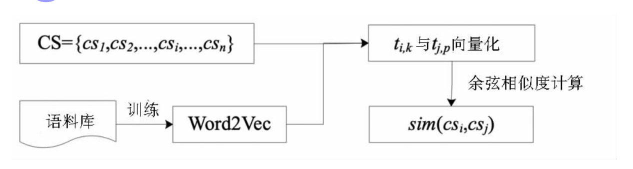


图1 语词对相似度计算过程

以表1所示由Word2Vec所构建的语词对打分矩阵为例，根据本文所选语料库训练的Word2Vec模型构建语词对打分矩阵，为保证更准确的打分效果，设定打分条件为$sim(t_{i,k},t_{k,p}) > 0.65$，此时，图1所示的打分矩阵中就包含了所有$sim(t_{i,k},t_{k,p}) > 0.65$的语词对，而所有$sim(t_{i,k},t_{k,p}) \leqslant 0.65$的语词对根据打分规则统一计为－0.05。当使用序列比对算法来度量两中文序列的相似性时，该语词对打分矩阵就为如何比较词语之间的相似性提供了一个参考，如比对过程中"国内"与"城市"比对在一起，G = 0.69分；"基于"与"推荐"比对在一起，G = －0.05分，直到比对结束，再综合两序列之间具有相似性的词语来判定两中文序列整体或局部上的相似关系，所构建的语词对打分矩阵越加准确和规范，序列比对算法的效果就越好，比对结果就更为准确，从而更好地度量两中文文本的相似度。

表1 语词对相似度打分矩阵

	基于	城市	知识图谱	推荐	现状	…	研究
基于	1.00	－0.05	－0.05	－0.05	0.69	…	0.71
国内	－0.05	0.69	－0.05	－0.05	0.74	…	－0.05
知识图谱	－0.05	－0.05	1.00	－0.05	－0.05	…	－0.05
…	…	…	…	…	…	…	…
现状	－0.05	0.66	－0.05	－0.05	1.00	…	0.72

2.3.2 全局比对算法

目前的相关研究对于如何将全局比对算法应用于中文文本相似度的计算已有一定的研究，该算法旨在从整体上分析两条中文序列的相似关系，即考虑两序列的总长，对两序列中所有的字符进行比对来寻找能使得全局相似性最大化的解。

为阐明传统的全局比对算法，以两中文文本"基于GIS的城市规划知识图谱的研究现状与趋势"和"基于国内医疗知识图谱的医生个性化推荐研究"进行全局比对为例，分词后得到cs_i = {*基于,GIS,城市,规划,知识,图谱,研究,现状,趋势*}，cs_j = {*基于,国内,医疗,知识,图谱,医生,个性化,推荐,研究*}，介于中文纷繁复杂的语词组合、复杂多变的中文文法以及中文表达"前轻后重"等特点，中文文本里更为重要的内容常常出现在后半部分，因此，cs_i与cs_j进行全局比对时从尾到头进行比对，比对过程如图2所示，比对过程中根据打分规则及表1所示打分矩阵进行比对打分，直到全部的词语比对结束为止，比对过程中会对打分进行累加以获得比对得分，动态规划寻找最终比对得分最高的那组解作为最优解。

比对完成后，图2所示比对得分为4.34的这组解为最优解，此时还需要从头到尾进行回溯，获取该最优解的比对路径并确保比对路径的准确性，最终，获得表2所示的结果(图2所示比对矩阵展示了cs_i与cs_j的比对过程，其中横向箭头与纵向箭头表示空位匹配，即某一序列的词与另一序列的空位符"－"匹配；斜箭头表示词语与词语的匹配，即表示完全匹配、相似匹配或错位匹配)。

	基于	GIS	城市	规划	知识	图谱	研究	现状	趋势	gap
基于	**4.34**	0.25	0.30	0.35	0.40	0.45	0.50	-0.55	-0.45	-0.45
国内	3.29	**3.34**	**3.39**	0.40	0.45	0.50	0.55	-0.50	-0.40	-0.40
医疗	2.55	2.60	2.65	**2.70**	0.50	0.55	0.60	-0.45	-0.35	-0.35
知识	2.55	2.60	2.65	2.70	**2.75**	0.60	0.65	-0.40	-0.30	-0.30
图谱	1.50	1.55	1.60	1.65	1.70	**1.75**	0.70	-0.35	-0.25	-0.25
医生	0.45	0.50	0.55	0.60	0.65	0.70	**0.75**	-0.30	-0.20	-0.20
个性化	0.50	0.55	0.60	0.65	0.70	0.75	**0.80**	-0.25	-0.15	-0.15
推荐	0.55	0.60	0.65	0.70	0.75	0.80	**0.85**	-0.20	-0.10	-0.10
研究	0.60	0.65	0.70	0.75	0.80	0.85	**0.9**	-0.15	-0.05	-0.05
gap	-0.45	-0.40	-0.35	-0.30	-0.25	-0.20	-0.15	**-0.10**	**-0.05**	**0.00**

图 2 cs_i 与 cs_j 的全局比对矩阵

表 2 cs_i 与 cs_j 全局比对的最优解比对路径

打分规则	完全匹配	空位匹配	相似匹配	错位匹配	完全匹配	完全匹配	空位匹配	空位匹配	空位匹配	完全匹配	空位匹配	空位匹配
cs_i	基于	GIS	城市	规划	知识	图谱	–	–	–	研究	现状	趋势
cs_j	基于	–	国内	医疗	知识	图谱	医生	个性化	推荐	研究	–	–
G	1	-0.05	0.69	-0.05	1.00	1.00	-0.05	-0.05	-0.05	1.00	-0.05	-0.05
S	4.34	3.34	3.39	2.70	2.75	1.75	0.75	0.80	0.85	0.90	-0.10	-0.05

全局比对结束后会获得序列长度一样的两条中文序列。cs_i 与 cs_j 的全局比对之前的序列长度(词语个数)分别为 $L_i=9$,$L_j=9$,全局比对结束后,由于空位匹配的情况在中文序列中插入了空位符"–"(空位符在比对中视作一个词语),使得 cs_i 与 cs_j 的最优解序列长度 $L_i=L_j=12$。基于最优解比对得分与序列长度,参考公式(1),两序列的相似度为 $sim(cs_i,cs_j)=4.34/12=0.362$。全局比对算法属于动态规划算法,其比对过程存在很多重复计算,在获得最优得分与最优比对路径的过程中相当于做了正比于比对矩阵 M 大小的 m×q 次计算,其时间复杂度为 O(n^2)。

$$sim(cs_i,cs_j)=\sum_{n=1}^{L}\frac{sim(t_{i,k},t_{j,p})}{L} \qquad 公式(1)$$

全局比对算法虽然能够较好地应用于内容差异较小的两文本的相似度计算,但在如下两个方面仍存在着很大的局限:

(1)比对内容差异较大的 cs_i 与 cs_j 时效果较差。全局比对算法对于全局上有较多相似之处的 cs_i 与 cs_j 具有不错的效果,但如果所比对的 cs_i 与 cs_j 只有少数词语存在相似关系,则很容易出现如表 3 所示的结果,即由于所示的两中文序列整体上的相似性较差,全局比对算法在寻找 cs_i 与 cs_j 全局上的相似性时会出现大量的空位匹配,这不仅降低了比对得分,而且增加了最优解下的序列长度,从而导致 $sim(cs_i,cs_j)$ 的效果与准确性均有所下降。

(2)比对序列长度差异较大的 cs_i 与 cs_j 时效果极差。与(1)原理相同,当所比对的 cs_i 与 cs_j 中有一方包含了更多的词语时,全局比对算法在递归寻找最优解的过程中需要遍历更多的词语,同时会插入更多的空位符来补全 cs_i 与 cs_j 的比对结果,此时,最优解的比对得分明显下降,最优解的序列长度也显著增加,导致全局比对算法的效果大打折扣。

基于上述,笔者应用局部比对算法来有效解决内容差异或长短差异较大的两本文进行比对的问题,以进一步提升序列比对算法的计算中文文本相似度的效果与准确性。

2.3.3 局部比对算法

全局比对算法旨在寻找 cs_i 与 cs_j 全局上的最优解,而局部比对算法则是寻找 cs_i 与 cs_j 局部上的最优解,局部比对在比对过程中所有 S 值小于 0 的分值都记为 0 而非负值,同时,比对结束后,回溯返回一个包含最大 S 值的子序列,而非完整的序列;局部比对是在整个比对过程中寻找中文两序列局部上 S 值最高的这一组解作为最优解,但 S 值所对应的解只包含两序列在局部上比对出的部分词语。以 cs_i = {基于,GIS,城市,规划,知识,图谱,研究,现状,趋势} 与 cs_j = {基于,国内,医疗,知识,图谱,医生,个性化,推荐,研究} 进行局部比对为例,如图 3 所示,在比对矩阵中,所有小于 0 的得分都被计为 0,从尾到头进行比对可得 S=3.59的这一组解为最优解,比对结束后,回溯出如表 3 所示的

最优解比对路径，参考公式(2)，可得 $sim(cs_i, cs_j) = 3.59/(9+9)2 = 0.399$。由于局部比对的最优解只包含 cs_i 与 cs_j 中的部分词语，而不涉及到完整的比对路径，此时局部比对最优解序列长度为 cs_i 与 cs_j 的初始长度求平均即 $(L_i + L_j)/2 = 9$。

	基于	GIS	城市	规划	知识	图谱	研究	现状	趋势	gap
基于	**3.59**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
国内	2.59	**2.59**	**2.64**	0.00	0.00	0.00	0.00	0.00	0.00	0.00
医疗	1.80	1.85	1.95	**1.95**	0.00	0.00	0.00	0.00	0.00	0.00
知识	1.80	1.85	1.90	1.95	**2.00**	0.00	0.00	0.00	0.00	0.00
图谱	0.75	0.80	0.85	0.90	0.95	**1.00**	0.00	0.00	0.00	0.00
医生	0.00	0.00	0.00	0.00	0.00	0.00	**0.00**	0.00	0.00	0.00
个性化	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**0.00**	0.00	0.00
推荐	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**0.00**	0.00
研究	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**0.00**
gap	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**0.00**

图 3　cs_i 与 cs_j 的局部比对矩阵

表 3　改进的局部比对算法的最优解比对路径

打分规则	完全匹配	空位匹配	相似匹配	错位匹配	完全匹配	完全匹配	错位匹配	错位匹配	错位匹配	空位匹配
cs_i	基于	GIS	城市	规划	知识	图谱	研究	现状	趋势	–
cs_j	基于	–	国内	医疗	知识	图谱	医生	个性化	推荐	研究
G	1.00	–0.05	0.69	–0.05	1.00	1.00	–0.05	–0.05	–0.05	–0.05
S	3.59	2.59	2.64	1.95	2.00	1.00	0.00	0.00	0.00	0.00

$$sim(cs_i, cs_j) = \sum_{n=1}^{L} \frac{sim(t_{i,k}, t_{j,p})}{(L_i + L_j)/2} \quad \text{公式(2)}$$

3 改进的中文序列比对算法

目前有关序列比对算法的研究大多是将全局比对算法应用于文本相似度的计算，而缺少对于局部比对算法的研究，同时，全局比对算法用于文本相似度的计算还存在着一定的局限性。因此，笔者在目前相关研究基础上，引入词性标注以更好地度量词语间的相似关系，从而提高全局比对算法的准确性，同时创新性地运用了局部比对算法来对全局比对算法进行优化。

改进的中文序列比对算法具体流程如图 4 所示，首先构建好需要进行比对的 CS，基于构建好的语词对打分矩阵和改进的打分规则对选择合适的序列比对算法来比对 cs_i 与 cs_j，最后，根据最优解的比对路径计算其相似度。改进后可选用的序列比对算法如下：

(1) 全局比对。适用于 cs_i 与 cs_j 词语个数的比值小于 2(长序列与短序列的语词个数比值)且全局相似性较好的 cs_i 与 cs_j 进行比对。

(2) 局部比对。适用于 cs_i 与 cs_j 词语个数的比值小于 2 且在内容上差异较大或词语间的相似度较低的 cs_i 与 cs_j 进行比对。

(3) 多次局部比对。适用于 cs_i 与 cs_j 词语个数的比值大于 2 的 cs_i 与 cs_j 进行比对。

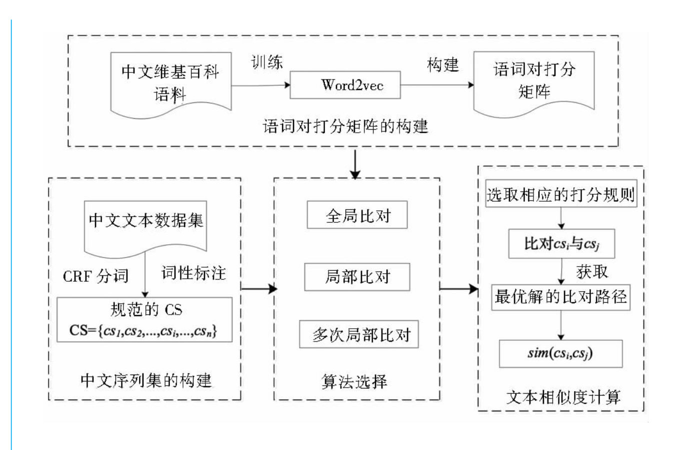


图 4　改进的中文序列比对算法流程

综上所述，当 cs_i 与 cs_j 词语个数的比值小于 2 时，可直接对这类序列进行全局比对，此时所获取的解中，在全局上具有较好相似性的 cs_i 与 cs_j 就能获得较高的相似度计算结果，然后，对那些相似计算结果较低的 cs_i 与 cs_j 使用局部比对算法，来寻找出那些局部上有较好相似关系的 cs_i 与 cs_j，从而提升这类序列的相似性计算结果。当 cs_i 与 cs_j 词语个数的比值大于 2 时，较长序列的词语数量会多出短序列很多，此时若进行全局比对或局部比对，则会出现大量的空位匹配而导致很差的相似度计算结果，考虑到较长序列中可能存在不

止一处与短序列有相似关系,因此对这类序列进行多次局部比对。

3.1 基于词性标注的全局比对算法优化

在使用全局比对算法比对 cs_i 与 cs_j 的过程中,词语之间的匹配打分是通过语词对打分矩阵来提供的,如"研究"与"研究","服务"与"服务"的相似度都为1,但这两个词在不同的语境中的用法和词性可能有所不同,如"研究"与"服务"两个词都能用做动名词(vn)、动词(v)以及名词(n),此时,若所比对上的两个词为"研究/v"与"研究/n",*sim*("研究/v","研究/n")应当不为1。因此笔者在使用CRF模型处理中文文本数据时,除了分词还进行了词性标注,即在比对打分的过程中充分考虑词性这一因素来改进打分规则,从而进一步提高全局比对算法的准确性。由于词性标注的结果很细,具体到机构、职业、职务等,比对前,根据实证数据,笔者对一些相同属性的词语进行了合并,如动名词(vn)、专有名词(nx、ng等)等合并为名词,最终改进的打分规则具体如下:

(1)完全匹配。完全匹配的两词语若词性相同,则打分1分;若词性不同,则扣除0.05分,打分 G = 1 − 0.05 = 0.95分。

(2)相似匹配。相似匹配的两词语若词性相同,则参考语词对打分矩阵直接打分 $G = sim(t_{i,k}, t_{k.p})$;若词性不同,则扣除0.05分,$G = sim(t_{i,k}, t_{k.p}) - 0.05$。

(3)错位匹配。错位匹配的两词语若词性相同,则奖励0.05分,G = −0.05 + 0.05 = 0分;若词性不同,则 G = −0.05分。

(4)空位匹配。空位匹配的两词语的打分统一为 G = −0.05。

以表4所示 cs_i 与 cs_j 的最优解比对路径为例,基于改进后的打分规则,比对过程中,第三组语词对的打分为 *sim*("城市","国内") − 0.05 = 0.64,第四组语词对的打分为 −0.05 + 0.05 = 0,以此来规范比对过程中对每一组匹配结果的打分,从而进一步提高全局比对算法应用于中文文本相似度计算的合理性与准确性。参考公式(1)计算可得 $sim(cs_i, cs_j) = 4.39/12 = 0.366$。

表4 基于词性标注的全局比对最优解比对路径

打分规则	完全匹配	空位匹配	相似匹配	错位匹配	完全匹配	完全匹配	空位匹配	空位匹配	空位匹配	完全匹配	空位匹配	空位匹配
cs_i	基于	GIS	城市	规划	知识	图谱	–	–	–	研究	现状	趋势
词性	p	n	n	n	n	n				n	n	n
cs_j	基于	–	国内	医疗	知识	图谱	医生	个性化	推荐	研究	–	–
词性	p		n	n	n	n	n	n	n	n		
G	1	−0.05	0.69	−0.05 +0.05	1.00	1.00	−0.05	−0.05	−0.05	1.00	−0.05	−0.05
S	4.39	3.39	3.44	3.75	2.75	1.75	0.75	0.80	0.85	0.90	−0.10	−0.05

3.2 利用局部比对算法改进全局比对算法

当 cs_i 与 cs_j 的内容差异较大时,全局比对算法并不适用,基于改进后的打分规则,笔者应用局部比对算法,来递归求解内容差异较大的两中文序列局部上的相似性而非全局上的相似性,以更好地度量与计算 cs_i 与 cs_j 的相似度。

以表5所示 cs_i 与 cs_j 的最优解比对路径为例,在该路径中,所有 S < 0 的值都被计为0,基于改进后的打分规则,比对过程中,错位匹配的四组词语由于词性相同 G = 0.00分,最终,参考公式(2)计算可得 $sim(cs_i, cs_j) = 3.64/9 = 0.404$。

表5 改进的局部比对算法的最优解比对路径

打分规则	完全匹配	空位匹配	相似匹配	错位匹配	完全匹配	完全匹配	错位匹配	错位匹配	错位匹配	空位匹配
cs_i	基于	GIS	城市	规划	知识	图谱	研究	现状	趋势	–
词性	p	n	n	n	n	n	n	n	n	
cs_j	基于	–	国内	医疗	知识	图谱	医生	个性化	推荐	研究
词性	p		n	n	n	n	n	n	n	n
G	1.00	−0.05	0.69	−0.05 +0.05	1.00	1.00	−0.05 +0.05	−0.05 +0.05	−0.05 +0.05	−0.05
S	3.64	2.64	2.69	2.00	2.00	1.00	0.00	0.00	0.00	0.00

3.3 多次局部比对——基于局部比对算法的改进

当 cs_i 与 cs_j 的序列长度差异较大时,全局比对算法已然难以适用,基于改进的打分规则与局部比对算法,当较长序列的词语数量是短序列的2倍及以上时,

首先对较长的中文序列进行切分，再分别与较短的中文序列进行局部比对，最后，综合全部的比对结果来计算 $sim(cs_i, cs_j)$。

以 cs_i =[科学知识，图谱，学科知识，服务，应用，探析] 与 cs_j =[通过，科学知识，图谱，人文，社科类，学科，自然科学，学科，具体，应用，实例，梳理，总结，科学知识，图谱，学科知识，服务，应用，特点] 为例，cs_i 与 cs_j 所含词语个数分别为 $L_i=6$ 与 $L_j=19$，首先，以 cs_i 的词语个数为基础对 cs_j 进行切分，将 cs_j 切分为三个序列 cs_{j1}，cs_{j2}，cs_{j3}，三个序列分别含有6、6、7个词；然后，参考语词对打分矩阵和改进的打分规则将 cs_i 依次与这三个序列进行局部比对；最终，综合三次的比对结果（见表6），参考公式（3）计算得到相似度 $sim(cs_i, cs_j) = (2.61+1.51+5.67)/[6\times2+6\times2+(6+7)]/2 = 0.529$。

表6 改进的多次局部比对算法的最优解比对路径

打分规则	空位匹配	完全匹配	完全匹配	错位匹配	错位匹配	相似匹配	空位匹配
cs_i	-	科学知识	图谱	学科知识	服务	应用	探析
词性		n	n	n	v	n	n
cs_{j1}	通过	科学知识	图谱	人文	社科类	学科	-
词性	p	n	n	n	n	n	
G	-0.05	1.00	1.00	-0.05	-0.05	0.71	-0.05
S	2.56	2.61	1.61	0.61	0.66	0.71	0.00
打分规则	错位匹配	错位匹配	错位匹配	相似匹配	相似匹配	错位匹配	
cs_i	科学知识	图谱	学科知识	服务	应用	探析	
词性	n	n	nz	v	n	n	
cs_{j2}	自然科学	学科	具体	应用	实例	梳理	
词性	n	n	ad	v	n	n	
G	-0.05+0.05	-0.05	-0.05	0.69	0.82	-0.05+0.05	
S	1.41	1.41	1.46	1.51	0.82	0.00	
打分规则	空位匹配	完全匹配	完全匹配	完全匹配	完全匹配	完全匹配	相似匹配
cs_i	-	科学知识	图谱	学科知识	服务	应用	探析
词性		n	n	n	v	n	n
cs_{j3}	总结	科学知识	图谱	学科知识	服务	应用	特点
词性	v	n	n	n	v	v	n
G	-0.05	1.00	1.00	1.00	1.00	1.00	0.67
S	5.62	5.67	4.67	3.67	2.67	1.67	0.67

$$sim(cs_i, cs_j) = \sum_{n=1}^{L} \frac{sim(t_{i,k}, t_{j,p})}{[(L_1 + L_2 \cdots + L_m) + L_i \times m]/2}$$

公式（3）

4 实验研究

本节针对采集的中文文本进行比对，并以具有代表性的比对结果来展示和分析不同序列比对算法的优势之处与算法的准确性。

4.1 中文文本数据采集与预处理

为检验所使用方法的实践价值与应用价值，本文选取了在线学术资源数据作为所要研究的中文文本数据集。鉴于CNKI具备丰富全面的文献资源、快速迅捷的检索窗口以及精准清晰的批量检索等优势，笔者导出了中国知网2020年1月-2020年8月检索主题为“知识图谱”的中文文献题名、关键词以及摘要作为在线学术资源数据。同时，考虑到训练词向量模型所需语料的规模、全面性以及时效性，笔者下载最新的中文维基百科语料库来训练Word2Vec。

针对在线学术资源数据集，以“知识图谱”为关键词检索出的部分文献，其题名与知识图谱无关，但其内容可能有关，因此仍保留这部分数据，最后对所有的在线学术资源数据集，使用HanLP的CRF模型依次进行分词、词性标注，最终，得到一个包含了747条数据的在线学术资源中文序列集OARCS，部分数据如表7所示，因摘要过长，表中仅显示部分文献题名数据。

针对训练语料，访问 https://dumps.wikimedia.org/zhwiki/latest/下载最新的维基百科语料XML文件，从该XML文件中抽取出所有的中文文本并使用Python中的OpenCC完成繁简转化，最终，再次使用HanLP的CRF模型进行分词，以供给后续训练Word2Vec使用。

表 7 在线学术资源中文序列集

cs_1	基于	领域	知识	图谱	生命	医学	学科知识	发现	探析	
词性	p	n	n	n	n	n	n	v	n	
cs_2	基于	大规模	开放	学术	图谱	研究	前沿	分析	框架	
词性	p	d	v	n	n	n	S	v	n	
…	…	…	…	…	…	…	…	…	…	…
cs_{747}	当代	中国	卓越	教师	研究	热点	知识	图谱	可视化	分析
词性	t	n	a	n	n	n	n	n	v	v

4.2 语词对打分矩阵的构建

语词对打分矩阵作为序列比对算法的核心基础，直接关系到序列比对算法的最终效果和准确性，因此笔者在运用 Word2Vec 来构建语词对打分矩阵的基础上，还对打分矩阵进行了一定的优化。

4.2.1 Word2Vec 的训练与语词对相似度计算

使用处理好的维基百科语料来训练 Word2Vec 的 Skip-Gram 模型。相关参数设定如下：词向量维度 Size =100，向量上下文距离 Windows =5，忽略语料中的最小词频 min_count =1。基于预处理好的 OARCS 以及训练好的 Word2Vec，进一步计算词语之间的相似度，对于 OARCS 中任意两条需要进行比对的中文序列 $cs_i=\{t_{i,1},t_{i,2},\cdots,t_{i,k},\cdots,t_{i,m}\}$ 与 $cs_j=\{t_{j,1},t_{j,2},\cdots,t_{j,p},\cdots,t_{j,q}\}$，运用 Word2Vec 计算两序列之间任意两词语 $t_{i,k}$ 与 $t_{j,p}$ 词向量的余弦相似度 $sim(t_{i,k},t_{j,p})$ 作为构建语词对打分矩阵的核心基础。

4.2.2 语词对打分矩阵的优化

中文序列的比对，介于不同条件下，对于序列比对算法的比对粒度、精细度、准确度的要求不同，要适当地调整打分矩阵。打分矩阵中语词对的相似度介于 0 −1.0 之间，如图 5 所示，当对于两文本进行比对的准确度要求较高时，则调高 λ 的值，令打分矩阵保留 $sim(t_{i,k},t_{j,p})>\lambda$ 的语词对供给比对打分所参考，而将 $sim(t_{i,k},t_{j,p})<\lambda$ 的词语对放入非打分词库保存起来。若下一次比对又一次调整了 λ，只需参考 λ 调整语词对打分矩阵和非打分词库即可。

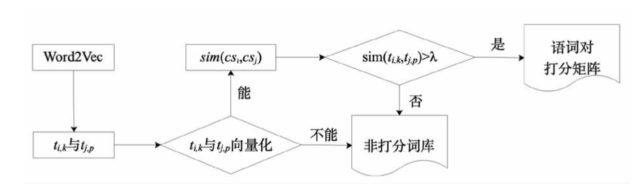


图 5 语词对打分矩阵的调整

在打分矩阵中，会存在部分重复的 $sim(t_{i,k},t_{j,p})$，例如当 cs_i 与 cs_j 中都含有“研究”“探究”两个词语，则计算语词对相似度后会出现 *sim*（‘研究’，‘探究’）= 0.80、*sim*（‘探究’，‘研究’）= 0.80，为避免这类语词对在打分矩阵中占用额外的空间，从而导致本文算法出现更高的时间复杂度和空间复杂度，则需要对这类结果进行去重再放入打分矩阵当中以提高序列比对算法的运行效率，最终，针对所要比对的在线学术资源中文序列构建语词对打分矩阵，取 λ =0.7，此时，在打分矩阵中保留所有 $sim(t_{i,k},t_{j,p})>0.7$ 的语词对，同时，将 $sim(t_{i,k},t_{j,p})\leqslant 0.7$ 的语词对放入非打分词库中，得到如表 8 所示的打分矩阵。

表 8 OARCS 语词对打分矩阵

	创新	方法	分析	规划	技术	领域	热点	…	学科
服务	0.72	−0.05	−0.05	0.72	0.72	−0.05	−0.05	…	−0.05
科技	0.85	−0.05	−0.05	−0.05	0.85	0.78	−0.05	…	0.74
前景	0.80	−0.05	0.73	0.71	0.74	0.74	0.75	…	−0.05
特点	0.73	0.75	−0.05	−0.05	0.73	0.74	0.71	…	−0.05
推断	−0.05	−0.05	0.79	−0.05	−0.05	−0.05	−0.05	…	−0.05
…	…	…	…	…	…	…	…	…	…
探析	0.72	−0.05	0.7	−0.05	−0.05	−0.05	−0.05	…	−0.05

4.3 在线学术资源中文序列比对

经过前述部分，此时已获得预处理好的 OARCS 以及构建好的语词对打分矩阵，基于改进的打分规则，对中文序列从尾到头进行比对，部分所要对比的中文序列如表 9 所示，将表中左侧的序列与右侧序列进行比对，即 ID 为 cs_1 与 ID 为 cs_2 的序列进行比对，ID 为 cs_3 与 ID 为 cs_4 的序列进行比对，依此类推，直到 cs_i 与 cs_j 比对结束。

赵登鹏，熊回香，田丰收，等.基于序列比对算法的中文文本相似度计算研究[J].图书情报工作,2021,65(11):101－112.

表 9　所要比对的中文序列(部分)

ID	在线学资源中文序列	ID	在线学术资源中文序列
cs_1	{高校/j，科研/n，管理/n，研究/n，热点/n，趋势/n，分析/n，基于/p，CiteSpace/nx，可视化/n，分析/n}	cs_2	{高校/j，科研/n，绩效评价/n，研究/n，热点/n，趋势/n，基于/p，可视化/v，知识/n，图谱/n，分析/v]}
cs_3	{基于/p，Citespace/nx，城市/n，热风/n，环境/n，研究/n，知识/n，图谱/n，分析/v}	cs_4	{基于/p，CiteSpace/nx，城市/n，生态/n，修复/v，研究/v，知识/n，图谱/n，分析/v}
cs_5	{基于/p，知识/n，图谱/n，构建/v，高职院/n，校内部/n，控制/n，体系/n，研究/n}	cs_6	{基于/p，交互式/n，可视化/v，领域/n，知识/n，图谱/n，构建/v，研究/n}
cs_7	{国际/n，人工/n，智能/n，伦理/n，研究/n，现状/n，发展趋势/l}	cs_8	{基于/p，知识/n，图谱/n，建筑/n，科学/n，工程/n，人工/n，智能/n，研究/n，趋势/n，分析/n}
cs_9	{科学知识/n，图谱/n，学科/n，精准/a，服务/n，中的/v，应用/v，探索/v}	cs_{10}	{通过/p，科学知识/n，图谱/n，人文/n，社科类/n，学科/n，自然科学/n，学科/n，具体/ad，应用/v，实例/n，梳理/n，总结/v，科学知识/n，图谱/n，学科知识/n，服务/v，应用/v，特点/n，指出/v，创新/v，形式/n，专利/n，图谱/n，学科/n，嵌入式/b，服务/n，技术/n，预见/n，科学知识/n，图谱/n，学科知识/n，服务/v，应用/n，前景/n，同时/c，分析/v，科学知识/n，图谱/n，学科知识/n，服务/v，运用/v，注意事项/n }
	…	…	…
cs_i	{国内/s，深度/n，学习/v，研究/v，知识/n，图谱/n }	cs_j	{文章/n，从/p，深度/n，学习/v，研究/n，现状/n，内涵/n，入手/v，利用/v，文献/n，计量法/n，有关/n，深度/n，学习/v，426/m，篇/q，文献/n，进行/v，论文/n，影响力/n，分析/n，对/p，排名/v，前/f，20%/m，文献/n，进行/v，文献/n，年代/n，期刊/n，发布/v，作者/n，分析/n，得出/v，深度/n，学习/v，初步/b，知识/n，图谱/n，以期/v，促进/v，深度/n，学习/n，技术/n，应用/n，发展/n }

4.4　实验结果及评价

为了凸显本文研究相较已有研究的优势之处，本文使用不同的序列比对算法比对并计算了表9所示中文序列的相似度，同时，设置了不同的语词对打分矩阵参数λ来进一步比较算法效果，不同算法所得结果见表10(其中，传统全局比对算法相似度计算结果不参考构建的打分矩阵与词性标注的结果)。显然，相较于传统的全局比对算法，本文方法基于Word2Vec构建的语词对打分矩阵与词性标注的结果所计算的文本相似度在整体上均有所提升。

细致比较表9所示中文序列，实际上，cs_1 与 cs_2，cs_3 与 cs_4 的比对，序列之间的内容差异与长短差异小(即序列之间全局相似性较好，且序列之间词语个数的倍数小于2)，使用全局比对算法的效果更加理想；cs_5 与 cs_6，cs_7 与 cs_8 的比对，序列之间的内容差异较大而长短差异较小(即序列之间局部相似性较好，同时，序列之间词语个数的倍数小于2)，此时使用局部比对算法的效果相比全局比对算法更为出色；cs_9 与 cs_{10}，cs_{11} 与 cs_{12} 的比对，序列之间长短差异很大(即序列之间词语个数的倍数大于2)，对这类序列的相似度计算，选用多次局部比对算法相比其他算法效果更好。

表 10　OARCS 在不同方法中的相似度计算结果

OARCS	OARCS	传统全局比对	改进的全局比对(λ=0.7)	改进的全局比对(λ=0)	局部比对(λ=0)	多次局部比对(λ=0)
cs_1	cs_2	0.596	0.608	0.740	0.729	0.729
cs_3	cs_4	0.777	0.777	0.901	0.896	0.896
cs_5	cs_6	0.388	0.490	0.665	0.704	0.704
cs_7	cs_8	0.213	0.328	0.382	0.533	0.533
cs_9	cs_{10}	0.072	0.108	0.107	0.225	0.493
…	..	…		…		…
cs_i	cs_j	0.064	0.079	0.079	0.200	0.584

为了更加直观地展现本文方法的效果，将表 10 做成图 6 所示折线图，前三组中文序列的比对，传统全局比对算法与改进的全局比对算法（$\lambda=0.7$）效果相当，主要是因为 $\lambda=0.7$ 时，调整后打分矩阵中满足打分条件 $sim(t_{i,k},t_{j,p})>\lambda$ 的相似语词就会大大减少，导致比对得分降低，最终，两种相似度计算结果的差异就较小。当调整为 $\lambda=0$ 时，打分矩阵中可供打分参考的相似词语显著增加，此时，本文方法的效果就有了较大的提升。

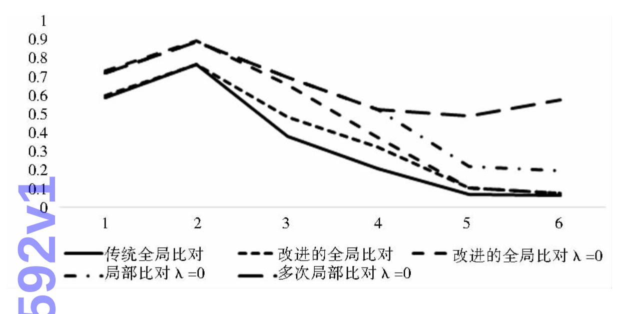


图 6 不同方法的相似度计算结果比较

从实证过程来看，构建打分矩阵的 Word2Vec 虽然能够很好地将训练其自身的语料以特征向量的形式表示出来，但这些词向量却无法表示文本原来的语词顺序，而本文所改进的全局比对算法和所应用的局部比对算法，均严格按照语词顺序来比对两个文本的相似之处，同时，参考 CRF 模型词性标注的结果与 Word2vec 构建的语词对打分矩阵，不仅考虑了词语之间的含义，也提升了本文方法的效果。传统的文本相似度计算方法将词语看作字符来进行比较缺乏对于词语之间含义与关联的考虑，本文方法使用 Word2Vec 有效解决了该问题，同时比对过程严格按照词语顺序进行，能够更好地度量两文本的相似关系。

改进的全局比对算法虽然能够较好地应用于在全局上具有较好相似关系的两中文序列的比对，但对于内容差异较大、长短差异较大的两中文序列的比对，效果较差，而本文中所应用的局部比对算法与多次局部比对算法则较好地解决了这一问题，使得序列比对算法能够更好地运用于中文文本相似度的计算。

5 结语

本文基于词性标注的结果与构建的语词对打分矩阵改进了全局比对算法，并应用局部比对算法来弥补了全局比对算法用于计算文本相似度的不足之处：①序列比对算法的效果非常依赖于研究前期对中文文本进行自然语言处理的效果，本文选用 HanLP 中对新词具有良好识别效果的 CRF 进行分词，保障了构建中文序列集合的规范性与准确性；②比对过程中对于匹配上的两词语进行词性判断，能够使得比对打分更加合理，从而提升序列比对算法的准确性，笔者实证使用的 CRF 模型融合了 HanLP 工具处理自然语言的核心技术，词性标注的结果也更加准确、细腻；③语词对打分矩阵的构建是序列比对算法得以良好应用于中文文本相似度计算的核心基础，虽然本文选取了通用语料来训练 Word2Vec，但是最新的维基百科语料库已经具备足够的覆盖广度和较大的语料规模，所构建的语词对打分矩阵有不错的参考价值和效果；④笔者合理运用局部比对算法，使得序列比对算法能够更好地应用于文本相似度计算的研究当中，这也为该算法用于语义挖掘、文本分类与聚类、个性化推荐、智能检索等奠定了一定的理论基础和实践基础。

经过大量的数据测试，基于图情领域出色的算法和工具，针对中文"特色"所改进的序列比对算法已能够较好应用到文本相似度比较当中。参考生物信息学领域对序列比对算法的研究，本文的研究已经为使用序列比对算法进行文本分类聚类打下了良好的基础，下一阶段，笔者将获取有效的中文数据，基于序列比对算法扎实的理论基础和丰富的研究成果，尝试对规范化的中文序列进行分类聚类，同时，结合图情领域的研究方法及算法工具进一步探索中文文本之间更深层次的关联与意义。

参考文献：

[1] MAHMOOD Q, QADIR M A, AFZAL M T. Application of cores to compute research papers similarity[J]. IEEE access, 2017, 5: 26124 – 26134.

[2] PARASCHIV I C, DASCALU M, TRAUSAN-MATU S, et al. Analyzing the semantic relatedness of paper abstracts: an application to the educational research field[C]//International conference on control systems and computer science. Bucharest: IEEE, 2015: 759 – 764.

[3] 黄文彬，车尚锟. 计算文本相似度的方法体系与应用分析[J]. 情报理论与实践，2019，42(11)：128 – 134.

[4] 章成志. 基于多层特征的中文字符串相似度计算模型[J]. 情报学报，2005，24(6)：696 – 701.

[5] 陈二静，姜恩波. 文本相似度计算方法研究综述[J]. 数据分析与知识发现，2017，1(6)：1 – 11.

[6] 李琳，李辉. 一种基于概念向量空间的文本相似度计算方法[J]. 数据分析与知识发现，2018，2(5)：48 – 58.

[7] 王春柳,杨永辉,邓霏,等. 文本相似度计算方法研究综述[J]. 情报科学,2019,37(3):158 -168.

[8] GOMAA W H, FAHMY A A. Short answer grading using string similarity and corpus-based similarity[J]. International journal of advanced computer science and applications,2012,3(11):114 -121.

[9] KADUPITIYA J, RANATHUNGA S, DIAS G. Short sentence similarity calculation using corpus-based and knowledge-based similarity measures[C]//Proceedings of the 26th international conference on computational linguistics. Osaka: The coling 2016 organizing committee,2016:44 -53.

[10] GOMAA W H, FAHMY A A. A survey of text similarity approaches[J]. International journal of computer applications,2013,68(13):13 -18.

[11] BOBADILLA J, ORTEGA F, HERNANDO A, et al. Improving collaborative filtering recommender system results and performance using genetic algorithms[J]. Knowledge-based systems,2011,24(8):1310 -1316.

[12] 文凤春,王邦菊,肖枝洪. 生物序列比对算法的研究现状[J]. 生物信息学,2010,8(1):66 -69.

[13] NEEDLEMAN S B, WUNSCH C D. A general method applicable to the search for similarities in the amino acid sequence of two proteins[J]. Journal of molecular biology,1970,48(3):443 -453.

[14] SMITH T F, WATERMAN M S, FITCH W M. Comparative biosequence metrics[J]. Journal of molecular evolution,1981,18(1):38 -46.

[15] FENG D F, DOOLITTLE R F. Progressive sequence alignment as a pre-requisite to correct phylogenetic trees[J]. Journal of molecular evolution,1987,25(4):351 -360.

[16] ALTSCHUL S F, GISH W, MILLER W, et al. Basic local alignment search tool[J]. Journal of molecular biology,1990,215(3):403 -410.

[17] EDDY S R. Multiple alignment using hidden markov models[J]. International conference on intelligent systems for molecular biology,1995(3):114 -120.

[18] THOMPSON J D, HIGGINS D G, GIBSON T J. Clustal w: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position specific gap penalties and weight matrix choice[J]. Nucleic acids research,1994,22(22):4673 -4680.

[19] NOTREDAME C HERINGA, J HIGGINS. T-coffee: a novel method for fast and accurate multiple sequence alignment[J]. Journal of molecular biology,2000,302(1):205 -217.

[20] LASSMANN T. Kalign 3: multiple sequence alignment of large data sets[J]. Bioinformatics,2019,36(6):1928 -1929.

[21] ZHANG C X, ZHENG W, MORTUZA S M, et al. Deepmsa: constructing deep multiple sequence alignment to improve contact prediction and fold-recognition for distant homology proteins[J]. Bioinformatics,2019,36(7):2105 -2112.

[22] LU R J, ZHAO X, LI J, et al. Genomic characterisation and epidemiology of 2019 novel coronavirus: implications for virus origins and receptor binding[J]. The lancet, 2020,395:565 -574.

[23] 徐硕,朱礼军,乔晓东,等. 基于双序列比对的中文术语语义相似度计算的新方法[J]. 情报学报,2010,29(4):701 -708.

[24] 王汀,徐天晟,冀付军. 基于数据场和全局序列比对的大规模中文关联数据模型[J]. 中文信息学报,2016,30(3):204 -212.

[25] 田久乐,赵蔚. 基于同义词词林的词语相似度计算方法[J]. 吉林大学学报(信息科学版),2010,28(6):602 -608.

[26] 熊回香,赵登鹏,卢晨凡. 基于词向量模型的中文序列比对研究[J]. 图书情报工作,2020,65(10):86 -98.

[27] SUTTON B C, MCCALLUM A. An introduction to conditional random fields[J]. Foundations & trends in machine learning,2010,4(4):267 -373.

[28] BUCHHOLZ S, MARSI E. Conll-x shared task on multilingual dependency parsing[C]//Tenth conference on computational natural language learning. New York: Association for computational linguistics,2006.

[29] JOULIN A, GRAVE E, BOJANOWSKI P, et al. Bag of tricks for efficient text classification[C]//Proceedings of the 15th conference of the European chapter of the Association for Computational Linguistics,2017:427 -431.

[30] DOZAT T, MANNING C D. Deep biaffine attention for neural dependency parsing[C]//The 5th international conference on learning representations. Puerto Rico: ICLR,2016:1 -8.

[31] DEVLIN J, CHANG M W, LEE K, et al. Bert: pre-training of deep bidirectional transformers for language understanding[EB/OL]/[2021 -01 -30]. http://arXiv:1810.04805.

[32] MIKOLOV T, SUTSKEVER I, CHEN K, et al. Distributed representations of words and phrases and their compositionality[J]. Advances in neural information processing systems,2013,26:3111 -3119.

[33] 郭思成,李纲,周华阳. 基于 Word2Vec 的医学知识组织系统互操作研究——以词表间语义映射为例[J]. 情报理论与实践,2019,42(9):160 -165.

作者贡献说明:

赵登鹏:算法设计与论文撰写;

熊回香:研究方向与方法提出;

田丰收:技术支持与指导;

李昕然:论文校对与修改。

Research on Chinese Text Similarity Calculation Based on Sequence Alignment Algorithm

Zhao Dengpeng[1] Xiong Huixiang[1] Tian Fengshou[2] Li Xinran[1]
[1] School of Information Management, Central China Normal University, Wuhan 430079
[2] Shandong Technology Center, GaoXunZhenYuan Education Technology Limited Company, Jinan 250000

Abstract: [**Purpose/significance**] Aiming at the application of sequence alignment algorithm in text similarity, the global alignment algorithm is improved and the accuracy of the algorithm is improved. At the same time, the local alignment algorithm is used to effectively solve the problem of comparing two texts with different content or with different length. [**Method/process**] First, the CRF model in HanLP was used to normalize the Chinese text data set of the online academic resources and constitute the Chinese sequence set. Then, Word2Vec model was trained with the latest Chinese Wikipedia corpus to construct the word pair scoring matrix. Finally, based on the scoring matrix and the improved scoring rules, the two Chinese sequences of global/local alignment were compared and the optimal solution of the alignment was obtained. The optimal solution was backtracked to obtain the alignment path of the optimal solution and the similarity of the two Chinese sequences was calculated. [**Result/conclusion**] The experiment results show that compared with the current research of global alignment algorithm, the method based on the results of the part-of-speech tagging and Word2Vec build words to further improve the global alignment score matrix algorithm and applied to the accuracy of computing text similarity of local alignment algorithm can effectively solve the content differences or differences in the length of two text comparing problems.

Keywords: CRF model part of speech tagging Word2Vec sequence alignment local alignment text similarity